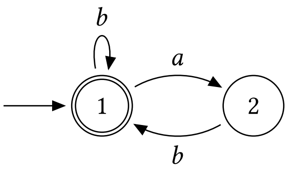
TD 17 — Automates finis

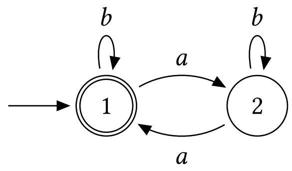
I. Automates déterministes

Exercice 1. Décrire par une phrase les langages reconnus par les automates suivants :

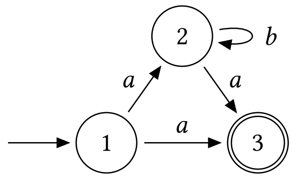
$\mathscr{A}_1$:



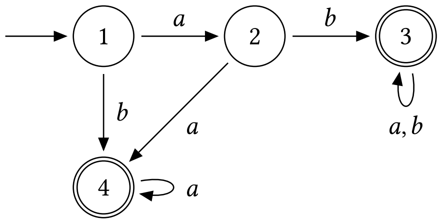
$\mathscr{A}_2$:



$\mathscr{A}_3$:



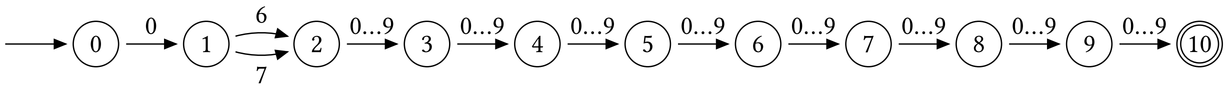
Exercice 2. Donner tous les mots de longueur au plus 4 acceptés par l'automate suivant :



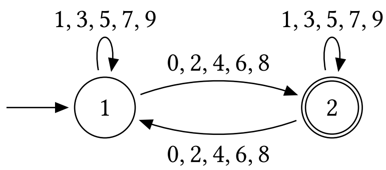
Exercice 3. Donnez une description formelle de l'automate précédent à partir de la définition vue en cours

Exercice 4. Décrire informellement les langages reconnus par les automates suivants.

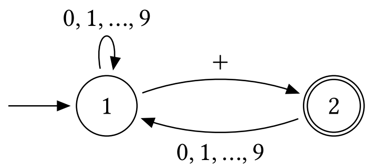
- Sur $\Sigma = \{0, 1, \dots, 9\}$, l'automate $\mathscr{A}_1$:



- Sur $\Sigma = \{0, 1, \dots, 9\}$, l'automate $\mathscr{A}_2$:



- Sur $\Sigma = \{0, 1, \dots, 9, +\}$, l'automate $\mathscr{A}_3$:

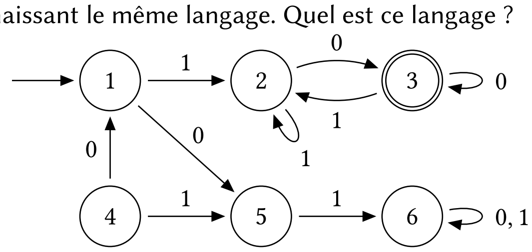


Exercice 5. Compléter les automates $\mathscr{A}_1$ et $\mathscr{A}_3$ de l'exercice précédent.

Exercice 6. Soit l'alphabet $\Sigma = \{a, b\}$. Dans chacun des cas suivants, donner l'automate reconnaissant le langage donné.

1. Le langage des mots contenant au moins un *a*.
2. Le langage des mots contenant exactement trois *a*.
3. Le langage des mots où tout *b* est suivi de deux *a*.
4. Le langage des mots ne contenant aucun facteur *aa* ou *bb*.
5. Le langage des mots ne contenant aucun facteur *aaa* ou *bbb*.

Exercice 7. Soit l'alphabet $\Sigma = \{0, 1\}$. Définir les états accessibles et ceux co-accessibles de l'automate ci-contre. En déduire l'automate émondé reconnaissant le même langage. Quel est ce langage ?



Exercice 8. Construire un automate fini déterministe sur l'alphabet $\Sigma = \{0, 1\}$, qui accepte les écritures binaires de nombres divisibles par 3. On considère que les mots sont lus de gauche à droite (en commençant par le bit de poids fort et en terminant par le bit de poids faible).

Exercice 9. Montrer qu'un automate fini déterministe reconnaissant le langage des mots sur $\Sigma = \{a, b\}$ dont la n-ième lettre en partant de la fin est un a possède au moins 2^n états.

Exercice 10. Soit $\mathscr{A}$ un automate fini et k un entier positif. Proposer un algorithme pour construire un mot de longueur k reconnu par $\mathscr{A}$ s'il en existe au moins un, ou signaler qu'il en existe pas. (Indice : utilisez la programmation dynamique). Donnez en la complexité.

II. Exercices plus difficiles

Exercice 11. Soit $m \in \Sigma^*$ où Σ est un alphabet quelconque. On veut construire un automate déterministe efficace (ayant un nombre minimal d'états) pour reconnaître le langage $\Sigma^* m$.

1. Dans le cas où $\Sigma = \{a, b\}$ et $m = aba$, construisez un automate déterministe à 4 états qui accepte le langage $\Sigma^* m$. (On pourra remarquer que aba admet 4 préfixes, et construire un état pour chaque préfixe.)
2. On peut généraliser la construction précédente en construisant l'automate des préfixes d'un mot m donné : un état par préfixe, et l'état p représente le fait que le plus grand préfixe de m qu'on vient de lire est p.

   Donnez une description formelle de l'automate $A_m = (Q, \Sigma, I, F, \delta)$ (pour m arbitraire) qui généralise la construction de la question précédente. (Pour construire δ, on pourra s'aider de la fonction σ_m qui a u associe le plus long suffixe de u qui est préfixe de m.)
3. Montrer par récurrence le lemme suivant : pour tout état $p \in Q$ et tout mot $u \in \Sigma^*$, en lisant u depuis l'état p, on arrive dans l'état $\sigma_m(pu)$.
4. Montrer que $L(A_m) = \Sigma^* m$.

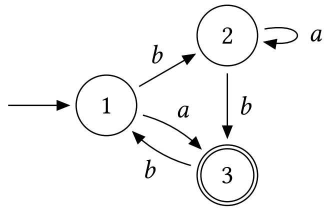
L'automate ainsi construit s'appelle aussi l'*automate des occurrences*, utilisé dans l'algorithme de Morris-Pratt, qui cherche un motif dans un texte. Celui-ci a été amélioré depuis en l'algorithme de Knuth-Morris-Pratt, qui n'a plus besoin de la construction de l'automate

Exercice 12 : *Lemme d'Arden.* On considère l'équation suivante, où A et B sont deux langages fixés tels que $\varepsilon \notin A$, et où X est un langage inconnu :

$$X = AX \mid B$$

1. Montrer que A^*B est solution de l'équation.
2. Soit L une solution de l'équation. Montrer que pour tout $n \in N$, on a $A^n B \subseteq L$. En déduire que $A^*B \subseteq L$.
3. Soit L une solution de l'équation, et $u \in L$ un mot. Montrer que $u \in A^*B$.
4. Conclure sur l'ensemble des solutions de l'équation $X = AX \mid B$

Considérons maintenant l'automate déterministe $\mathscr{A} = (Q, \Sigma, q_0, F, \delta)$ suivant :



5. Pour tout $i \in Q$, on note L_i le langage des mots qui sont l'étiquette d'un chemin depuis i vers un état acceptant. Écrire un système de 3 équations de langages, faisant intervenir les L_i, et les langages réduits à une lettre de Σ.
6. Résoudre le système d'équations précédent. En déduire une expression régulière qui engendre le langage accepté par l'automate.

Exercice 13 : *Système de transition déterministe.* On appelle *système de transition déterministe* un triplet $\mathscr{A} = (Q, \Sigma, \delta)$, obtenu en considérant un automate déterministe sans état initial ni état acceptant. Pour tout couple d'états $p, q \in Q$, on note $L_{p,q}$ le langage des mots qui sont l'étiquette d'un chemin de p à q.

1. Justifier que, pour un système de transition $\mathscr{A}$ et deux états p, q arbitraires, le langage $L_{p,q}$ est toujours régulier.

Dans la suite, on considère un langage K arbitraire, et on lui associe le langage $\sqrt{K} = \{u \in \Sigma^* \mid uu \in K\}$.

2. Supposons que K est reconnaissable par un automate déterministe $\mathscr{A} = (Q, \Sigma, q_0, F, \delta)$, et notons

$$L = \bigcup_{q_f \in F} \bigcup_{q \in Q} \left(L_{q_0,q} \cap L_{q,q_f} \right)$$

   Montrer que $L = \sqrt{K}$.
3. En déduire que si K est régulier, alors $\sqrt{K}$ est régulier.